REMARKS

The examiner rejected claims 1, 17 and 22 under 35 U.S.C. §101 on the ground that the claimed invention is directed to non-statutory subject matter. Specifically, the examiner stated:

> 3. As to claims, 1,17,22, although applicant recites a method operating on a processor, the structural elements are not being reflected into the claim body. Therefore, the processor is read as a general arrangement of element. Although applicant further recites executing the branch instruction including a first token specifying the number of instructions that are after the branch and before performing the branch and a second token to cause the processor to prefetch an instruction for the branch taken rather than the sequential instruction if the first token specifies zero or one instruction to execute after the branch , prior to performing an evaluation of a branch condition, it is read as an intended result. The reason is that no actual performance of branch, nor the actual prefetch has been taken place. Therefore, it is non-statutory.
>
> 4. As to claim 17,althought claim 17 reciting executing a branch instruction that causes a branch operation in instruction stream deferring performance of branch before performing branch, and evaluating the second token if the first token specifies zero or one instruction, no final result can be found. No actual prefetch has been taken place. Based on broadest interpretation, the "executing a branch instruction that causes..." is an intended use. Therefore, non-statutory. (Office Action, pages 2-3)

While applicant disputes the examiner's contentions that the independent claims recite intended results, to expedite prosecution of the above-identified application applicant amended claim 1 to recite the feature of retrieving, based, at least in part, on the second token, one of the instruction for the "branch taken" condition and the next sequential instruction. Thus, applicant's amended independent claim 1 recites performance of an instruction retrieval. Applicant similarly amended independent claims 17 and 22. Applicant further amended independent claim 17 to correct a typographical error.

The examiner also stated:

> 5. As to claim 22, claim 22 additionally disclosed decode logic. However, logic is an abstract idea. Although the preamble recites "processor", no components of processor have been reflected into the claim body.

To expedite prosecution of the above-identified application, applicant amended independent claim 22 to recite that the decode logic includes instruction decoder and a program counter, and to further recite that the decode logic is configured to perform the operations recited in claim 22. Support for this amendment is provided, for example, at page 4, lines 16-20 of the

application. Applicant notes that the instruction decoder and program counter of the decode logic are tangible items and not abstract ideas.

The examiner rejected claims 1, 3, 4, 6-8, 11-13, 17 and 19-22 under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,517,628 to Morrison et al. in view of U.S. Patent No. 4,606,025 to Peters et al. Further, the examiner rejected claims 24-26 under 35 U.S.C. §103(a) as being unpatentable over Morrison in view of Peters, and further in view of U.S. Patent No. 5,923,872 to Chrysos.

Specifically, with respect to independent claim 1, the examiner stated:

> 7. As to claims 1,4,12, 13,17, 22, Morrison taught at least executing a branch instruction of an instruction stream with a branch instruction including a first token specifying an amount of delay time, or deferring the performance (see delay field in col.43, lines 6, 17-26, see the amount of time that can be continued after receipt of branch in col.44, lines 61-67, col. 45, lines 1-12) that are after the branch instruction to execute before performing the branch operation and a second token (see condition code field in col.43, lines 17-26, col.44, lines 51-55) that specifies a branch guess operation (branch taken) to prefetch an instruction for the branch taken rather a next sequential instruction if the first token specified zero or one instruction to execute after the branch (see the non-zero and zero value of the delay field in col.45, lines 1-34, see the basic block as the sequential instructions). (Office Action, pages 3-4)

Applicant disagrees with the examiner's contentions.

Applicant's independent claim 1 recites "executing a branch instruction in execution of an instruction stream with the branch instruction including ... a second token that specifies a branch guess operation to cause the processor, if the first token specifies zero or one instructions to execute after the branch, to prefetch, prior to performing an evaluation of a branch condition associated with the branch instruction, an instruction for the "branch taken" condition rather than a next sequential instruction." Thus, applicant's branch instruction includes a second token that specifies a branch guess operation.

Morrison describes a computer with a condition code register file (Abstract). Morrison explains:

> **Also, the present invention makes extensive use of delayed branching. In order to guarantee program correctness, when a branch has executed and its effects are propagated in the system, all instructions that are within the procedural domain of the given branch must have been executed or be in the process of being executed as discussed with the example of Table 6. In other words, the changing of the next instruction pointer (in response to the branch) must take place after the current firing time has been updated to point to the firing time that would have followed the last (temporally**

> **executed) instruction governed by this branch. Hence, in the example of Table 6, instruction I5 at firing time T17 is delayed until the completion of T18 which is the last firing time for this basic block. The instruction time for the next basic block is then T19.**
>
> **...**
>
> **The branch instruction is delivered over bus 1546 from the PIQ bus interface unit 1544 into the instruction register 1900 of the BEU 1548. In FIG. 19 the fields of the instruction register 1900 are designated as: FETCH/ENABLE, CONDITION CODE ADDRESS, OP CODE, DELAY FIELD, and TARGET ADDRESS. The instruction register 1900 is connected over lines 1910a and 1910b to a condition code access unit 1920, to an evaluation unit 1930 over lines 1910c, a delay unit 1940 over lines 1910d, and to a next instruction interface 1950 over lines 1910e. Once an instruction has been issued to BEU 1548 from the PIQ bus interface 1544, instruction fetching must be held up until the value in the delay field has been determined. This value is measured relative to the receipt of the branch by the BEU, i.e. stage 1. If there are no instructions that may be overlapped with this branch, this field value is zero. In this case, instruction fetching is held up until the outcome of the branch has been determined. If this field is non-zero, instruction fetching may continue for a number of firing times given by the value in this field. (Col. 42, line 52, to col. 43, line 26)**

While Morrison's instructions include a delay field (see also FIG. 19), at no point does Morrison describe that any of its instructions include a field, or a token, that specifies a branch guess operation.

The examiner referred to Morrison's column 45, lines 1-34, in support of the contention that Morrison discloses a second token specifying a branch guess operation. The passage referred to by the examiner provides:

> **The delay unit 1940 determines the amount of time that instruction fetching can be continued after the receipt of a branch by the BEU. Previously, it has been described that when a branch is received by the BEU, instruction fetching continues for one more cycle and then stops. The instructions fetched during this cycle are held up from entering the PIQ 1544 until the length of the delay field has been determined. For example, if the delay field is zero (implying that the branch is to be executed immediately), these instructions must be withheld from the PIQ until it is determined whether or not these are the right instructions to be fetched. Otherwise, (the delay field is non-zero), the instructions would be gated into the PIQ as soon as the delay value was determined to be non-zero. The length of the delay is obtained from DELAY field of the instruction register 1900 and receives clock impulses from the context control 1518 over lines 1549a. The delay unit 1940 decrements the value of the delay with the clock pulses and when fully decremented the interface unit 1950 becomes enabled.**
>
> **Hence, in the discussion of Table 6, instruction I5 is assigned to firing time T17 but is delayed until firing time T18. During the delay time, the interface 1950 signals the instruction cache control 1518 over line 1549b to continue to fetch instructions to finish the current basic block. When**

enabled, the interface unit 1950 delivers the next address (i.e. the branch execution) for the next basic block into the instruction cache control 1518 over lines 1549b.

In summary and for the example on Table 6, the branch instruction I5 is loaded into the instruction register 1900 during time T17. However, a delay of one firing time (DELAY) is also loaded into the instruction register 1900 as the branch instruction cannot be executed until the last instruction I3 is processed during time T18. Hence, when the instruction I5 is loaded, the branch contained in the TARGET ADDRESS to the next basic block does not take place until the completion of time T18. In the meantime, the next instruction interface 1950 issues instructions to the context control 1518 to continue processing the stream of instructions in the basic block. Upon the expiration of the delay, the interface 1950 is enabled, and the branch is executed by delivering the address of the next basic block to the context control 1518. (Emphasis added, col. 44, line 61, to col. 45, line 34)

Thus, all Morrison says is that in the event that a branch instruction is to be executed without delay, subsequent instructions are withheld from the PIQ (processor instruction queue) until the branch condition is evaluated. Contrary to the examiner's contentions, there is no indication in this passage that any instruction, including Morrison's branch instruction, includes a branch guess token on the basis of which a processor determines whether to fetch instructions from the branch address, or to fetch a sequential instruction following the branch instruction.

Accordingly Morrison fails to disclose or suggest at least the features of "executing a branch instruction in execution of an instruction stream with the branch instruction including ... a second token that specifies a branch guess operation to cause the processor, if the first token specifies zero or one instructions to execute after the branch, to prefetch, prior to performing an evaluation of a branch condition associated with the branch instruction, an instruction for the "branch taken" condition rather than a next sequential instruction," as required by applicant's independent claim 1.

Peters describes a system for automatically testing a plurality of memory arrays on selected memory array testers (Abstract). At no point does Peters describe a branch instruction that includes a token specifying a branch guess operation. Accordingly, Peters too fails to disclose or suggest at least the features of "executing a branch instruction in execution of an instruction stream with the branch instruction including ... a second token that specifies a branch guess operation to cause the processor, if the first token specifies zero or one instructions to execute after the branch, to prefetch, prior to performing an evaluation of a branch condition

associated with the branch instruction, an instruction for the "branch taken" condition rather than a next sequential instruction," as required by applicant's independent claim 1.

Because neither Morrison nor Peters discloses or suggests, alone or in combination, at least the features of "executing a branch instruction in execution of an instruction stream with the branch instruction including ... a second token that specifies a branch guess operation to cause the processor, if the first token specifies zero or one instructions to execute after the branch, to prefetch, prior to performing an evaluation of a branch condition associated with the branch instruction, an instruction for the "branch taken" condition rather than a next sequential instruction," applicant's independent claim 1, and the claims depending from it, are patentable over the cited art.

Applicant's independent claims 17 and 22 recite "evaluating a second token that specifies a branch guess operation which causes the processor, if the first token specifies zero or one instructions to execute after the branch, to prefetch, prior to performing an evaluation of a branch condition associated with the branch instruction, an instruction for the "branch taken" condition rather than a next sequential instruction," or similar language. For reasons similar to those provided with respect to independent claim 1, at least these features are not disclosed by the cited art. Applicant's independent claims 17 and 22, and the respective claims depending from them, are therefore patentable over the cited art.

It is believed that all the rejections and/or objections raised by the examiner have been addressed.

In view of the foregoing, applicant respectfully submits that the application is in condition for allowance and such action is respectfully requested at the examiner's earliest convenience.

All of the dependent claims are patentable for at least the reasons for which the claims on which they depend are patentable.

Canceled claims, if any, have been canceled without prejudice or disclaimer.

Any circumstance in which the applicant has (a) addressed certain comments of the examiner does not mean that the applicant concedes other comments of the examiner, (b) made arguments for the patentability of some claims does not mean that there are not other good reasons for patentability of those claims and other claims, or (c) amended or canceled a claim

does not mean that the applicant concedes any of the examiner's positions with respect to that claim or other claims.

Enclosed is a Request for Continued Examination and a Petition for One Month Extension of Time. The fees in the amount of $810 and $120 are being paid concurrently on the Electronic Filing System (EFS) by way of Deposit Account authorization. Please apply any other required fees to deposit account 06-1050, referencing the attorney docket number shown above.

Respectfully submitted,

Date: Oct. 4, 2007

Ido Rabinovitch
Attorney for Intel Corporation
Reg. No. L0080

PTO Customer No. 20985
Fish & Richardson P.C.
Telephone: (617) 542-5070
Facsimile: (617) 542-8906

21751725.doc